The Thirty-First International Florida
Artificial Intelligence Research Society Conference (FLAIRS-31)

A Reading Recommendation System for ESL Learners Based on Linguistic Features

Mohamed Zakaria Kurdi
Department of Computer Science, Lynchburg College

Abstract

This paper presents a reading recommendation system based on morpho-phonological, lexical, and syntactic features reflecting both textual complexity and the learner's linguistic proficiency. The goal of this system is to optimize the reading process of ESL learners by proposing the fittest text to their needs given their incrementally built profile (weighted history of read texts). Fifteen features out of an initial pool of 90 candidates were selected. A corpus of 5052 texts of different levels was collected and used to build the system. To make the system more adaptive, a Progress Rate (PRate) measure was also proposed and integrated into the search process. Finally, the evaluation of the system showed positive results.

Introduction

The goal of this paper is to present an adaptive system dedicated to optimizing the reading experience of English as Second Language (ESL) learners. This system is based on modeling the text difficulty and the learner's proficiency on key linguistic features that are known to play an important role in text understanding. The system presented here aims, in other words, at transforming raw textual material collected from the web into an educational material.

Corpus

I collected a corpus of 5052 texts of English from several free professional websites about ESL[1]. The texts provided in these websites are organized by three levels of difficulty: 1, 2 and 3. These levels correspond respectively to A2, B1, and B2 in the Common European Framework of Reference for Languages (see (Council of Europe, 2011) for more information about this framework). 4964 texts in the corpus come from *news in level* website.

[1] I collected the texts from the following websites: http://www.newsinlevels.com/home/productscbm_550428/50/ http://learnenglishteens.britishcouncil.org/study-break/easy-reading - http://linguapress.com/inter.htm http://www.ngllife.com/content/reading-texts-word and http://www.fortheteachers.org/Reading_Resources/

Architecture of the system

The system is designed to build incrementally a user profile based on the previous interactions. As shown in Figure 1, the processing is done following two main modes: training and learning.

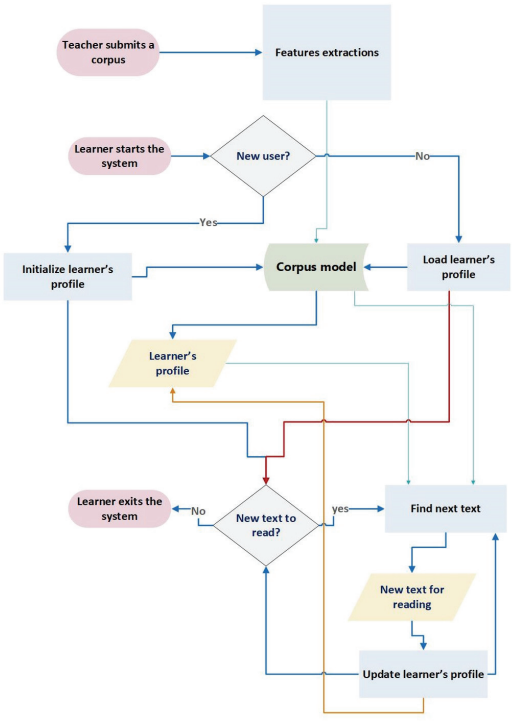

Figure 1. Flowchart of the system

In the training mode, a teacher provides the system with a corpus (set of texts) to train the learners on. The outcome of this step is the corpus model which is made of a set of vectors representing the key linguistic features of every text.

In the learning mode, the learner starts by choosing an ID and provides his English level. The level is obtained from an online test[2] and is used to initialize the learner's profile.

[2] http://www.cambridgeenglish.org/test-your-english/general-english/

Based on the initial profile, the system proposes the first text to read. Like with all proposed texts, the user provides a self-assessment about his understanding of the text. This helps to weigh the vector of the text's features that will be stored in the user profile.

Building the corpus model

The current reading recommendation system is based on characterizing the complexity of texts with a set of linguistic features automatically extracted from the texts. I started with an initial feature set of 90 linguistic features. These features cover all the areas of linguistic complexity such as phonology, morphology, lexicon, syntax, and discourse (Kurdi, 2017).

Among the initial pool of 90 features, I selected 15 features who satisfy three different, but complementary, criteria.

The chosen features can be grouped into three linguistic subdomains: morpho-phonology (1-5), lexicon (6-8), and syntax (9-15). Below is a brief description of some of these groups of features.

The mean number of phonemes per word describes the phonemic complexity of a word. The idea here is that the longer a word, the harder it is from a phonological point of view for a language learner. The extraction of this feature relies on the CMU pronouncing dictionary[3]. On the other hand, the mean number of suffixes per word accounts for the morphological complexity of words.

Lexical information is central in the language learning process. Among the measures for lexical diversity, I used GTTR. It is one of many attempts to find a mathematical transformation of the classic Type/Token Ratio (TTR),[4] which is known for its sensitivity to the size of the text (Guiraud, 1960). Furthermore, to account for the lexical sophistication, I adopted the Verb Sophistication Measure (VSM) proposed by (Harley and King, 1989). VSM is calculated as the ratio of the number of sophisticated verbs to the total number of verbs (see equation 1).

$$VSM=\frac{\#\text{sophisticated verbs}}{\#\text{ verbs}} \quad \text{(eq. 1)}$$

Practically, are considered as sophisticated the verbs whose frequency rank is higher than 200[5] in the McMillan English Dictionary, which contains a list of the 330 most frequent verbs[6]. To find the uninflected form of a verb, I used the verb conjugation module provided within the Pattern.en toolbox[7].

Beyond their lexical dimension, discourse connectors yield an important insight about the complexity of information structuring within the text. I calculated this factor as the number of lexical connectors divided by the number of words within the text.

Syntactic features are the most numerous and cover the different aspects of sentence syntax. For example, the mean number of phrases and the mean length of phrases cover the extent of a sentence, which is a source of difficulty; the longer the sentence the harder it is to understand. On the other hand, the percentage of inverted declarative sentences[8], the number of subordination per sentence, the mean phrase coordination per phrase, the mean height of parse trees, and the percentage of complex T-units per T-units are all used to measure different aspects of the complexity of syntactic dependencies within the text (see Table 1).

	Measure	Cramer's V	$\mu(\delta(\varphi))$
1	Mean number of phonemes per word	0.456	0.20
2	Mean number of suffixes per word	0.484	0.04
3	% simple present	0.615	-0.25
4	% simple past	0.596	0.17
5	% present perfect	0.391	0.028
6	GTTR	0.362	129.74
7	VSM	0.258	0.02
8	Percentage of discourse connectors	0.325	0.009
9	Mean number of phrases per sentence	0.881	4.34
10	Mean lengths of phrases	0.790	1.21
11	% Inverted declarative sentence	0.90	0.002
12	Number of subordination per sentence	0.626	0.30
13	Mean phrase coordination	0.463	0.11
14	Mean height of parse trees	0.818	2.32
15	% complex T-units per T-units[9]	0.673	0.28

Table 1. Effect size (Cramer's V) and the mean delta of the adopted features. The p value of the chi square test is <0.001 in all the cases

Given that these features are used to measure not only the linguistic complexity of the texts but also the reading performance of the learner, it is important to measure the differences of their values between the levels for every feature. I call this the mean delta of the feature. It is calculated according to equation 2.

[3] http://www.speech.cs.cmu.edu/cgi-bin/cmudict?in=manor&stress=-s

[4] It is calculated as: # words types (NWT)*100/# words. See (Ure, 1971) for more details)

[5] 200 being empirically defined threshold.

[6] http://www.acme2k.co.uk/acme/3star%20verbs.htm

[7] http://www.clips.ua.ac.be/pages/pattern-en

[8] Where the subject follows the tensed verb or modal.

[9] T-unit is defined as the shortest grammatically allowable sentence into which writing can be split for more details see: https://en.wikipedia.org/wiki/T-unit

$$\mu(\delta(\varphi_i)) = \frac{\mu(\varphi_{i,2}) - \mu(\varphi_{i,1}) + \mu(\varphi_{i,3}) - \mu(\varphi_{i,2})}{2} \text{ (eq. 2)}$$

In equation 2, $\varphi_{i,k}$ is a feature i of level k. For instance, for the feature Percentage of Complex T-units per T-units (PCTT), I calculated the value of the delta as follows. First, I calculated the mean value for the levels one, two, and three, which are respectively: 0.18, 0.53, and 0.75. When we plug these values into equation 2, we get the following:

$\mu(\delta(\text{PCTT})) = \frac{0.53 - 0.18 + 0.75 - 0.53}{2} = 0.285$

The values of the delta of every feature are provided in Table 2. As we can see, these values vary considerably between the features. Although most are positive, only the percentage of simple present is negative. This is due to the fact that beginners' texts tend to rely more on the simple present than advanced tenses.

Building the learner's profile

At the beginning, the learner takes a pretest to know his English level. Given this level, an initial profile is attributed to the learner: three random text vectors, with the same proficiency level, are added to the initially empty profile.

When the learner reads a text, the vector of the text is weighted based on the self-attributed score. This vector is then added to the learner's profile. The self-assessment is based on Likert scale with the following five possibilities (see Table 2).

Assessment	1	2	3	4	5
Interpretation	too easy	easy	about right	hard	too hard
Weight	-0.05	-0.02	0	0.02	0.05

Table 2 Feature weights equivalent to text grades

The weighting is done separately for every feature because every feature has its own value of $\mu(\delta(\varphi))$ as we saw in Table 2.

Besides, it is well-known that every learner has his or her own learning pace depending on a wide number of interrelated factors such as academic ability (commonly referred to as intelligence), gender, learning style, etc.

Given that the features cover 3 major areas of language proficiency, it is also fair to assume that the speed of progress is not necessarily the same in all the areas of the language as a learner can, for instance, make faster progress with morphology but slower progress with lexicon or vice versa. To take this aspect of learning into account, I propose to integrate a Progress Rate (PRate) factor into the learner's model (see equation 3). Three different values of PRate per user are calculated based on all the vectors within the learner's profile. This means that we have a distinct value for each of the subdomains (D_k) of proficiency: morpho-phonology, lexicon, and syntax.

$$\text{PRate}(D_k) = \text{URate}(D_k) - \text{CRate}(D_k). \text{ (eq. 3)}$$

As we can see in equation 3, the PRate is calculated based on two main terms the URate and the CRate.

The URate (User Rate) is calculated as the sum of the features of a given subdomain. Given that the values of some syntactic features are not percentages, they are excluded from the calculation of the syntax score, for harmonization reasons. This score will, therefore, be limited to the following four features: 11, 12, 13, and 15. As depicted in equation 4 and Figure 2, the URate is calculated as the sum of the deltas of the features of the domain divided by the number of items in the user profile h.

$$\text{URate}(D_k) = \frac{\sum_{i=2}^{h} \left(\sum_{j=b_k}^{n_k} (\varphi_{j,i}) - \sum_{j=b_k}^{n_k} (\varphi_{j,i-1}) \right)}{h} \text{ (eq.4)}$$

In equation 4, D_k represents the linguistic subdomain, n_k represents the upper index of the features in the domain D_k while b_k represents the lower index. For more details, see the example provided in Figure 2 where the user has read 4 texts.

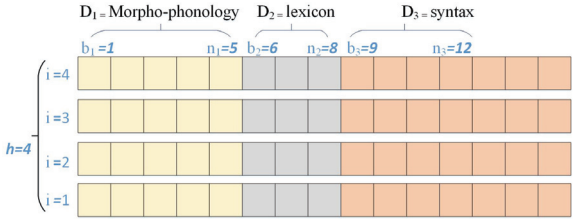

Figure 2 Example of URate variables for a user who read 4 texts

Then Corpus Rate (CRate) is calculated in a similar way to the URate (see equation 5). Except that here, I calculate this value offline for the entire corpus.

$$\text{CRate}(D_k) = \frac{\sum_{i=1}^{c} \left(\sum_{j=b_k}^{n_k} (\varphi_{j,k}) - \sum_{j=b_k}^{n_k} (\varphi_{j,k-1}) \right)}{c} \text{ (eq. 5)}$$

Note that in equation 5, c stands for the number of texts in the corpus.

Hence, PRate can be positive zero or negative as some learners exceed in their progress pace the natural differences that are in the corpus domain while other can go slower than it. Note that this rate is not a universal cognitive measure. It is just a way to decide the progress pace of difficulty given the learner's progress on a specific corpus. Obviously, the larger the corpus, the more accurate is the PRate. The PRate is initialized to zero for a new learner and then it gets incremented or decremented gradually.

Finding the fittest text

Suppose that $T = \{t_1, t_2, \ldots t_n\}$ is the learner sequence of read texts and that $C = \{c_1, c_2, \ldots c_k\}$ is the set of texts that are available in the corpus and that have not been read by the learner yet. I select the next text t_m such that the dis-

tance between T and t_m is optimal. As discussed in (Zampa and Lemaire, 2002) there are two different possible interpretations of the word *optimal*. If we search C for the nearest text to T, in this case, the system will select the text with the minimal differences from the learner's profile. This leads to a very slow progress and may not be very effective in terms of learning speed. On the other hand, the other extreme would be to propose the farthest text. Here, the system will select the text with a maximal distance from the learner's profile. This will speed up the progress, but most likely, it will not be optimal from a learning point of view as the suggested texts will be too hard for the learner.

To solve this problem, the fittest text is calculated as follows. First, I calculate the target vector. The target vector is a vector where every feature is the mean of the three previous values of the same feature within the learner's profile. A window of the last three vectors is used to take into consideration the recent progress of the learner, not the overall path stored in T that is necessarily representative of the current learner's level. To take into consideration the progress of the learner, each value within this vector is then weighted by multiplying it by the corresponding Prate. Finally, using the cosine distance formula (see equation 6), the distances between the weighted target vector and all the non-read texts within the corpus is carried out. The text with the shortest distance is selected.

$$\text{Similarity (A, B)} = \cos(\theta) = \frac{A.B}{||A||_2||B||_2} \quad \frac{\sum_{i=1}^{n}(A_i B_i)}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \text{(eq. 6)}$$

In equation 6, A_i and B_i are respectively components of the feature vectors A and B.

Evaluation

To provide an assessment of the performance of the system, an evaluation was carried out with five adults who are non-native English speakers with English levels ranging from A2 to B2 (levels covered by the system). All the subjects have a university degree or are college students. The subjects' experience with the system was evaluated with a questionnaire covering four basic facts. Each of these facts was evaluated by the subjects using Likert scaled grades where one is equivalent to *never* and five corresponds to *always*. Table 3 shows the questions and the means of the answers provided by the subjects.

As we can see in Table 3, where I normalized the results to the scale 0-100, the system does not require any specific technical skills to operate. Most importantly, it seems that the systems' users were generally satisfied with the level of the suggested texts. Hence, their overall evaluation of their experience was positive.

Facts	Evaluations' Means
It was clear what I had to do at each step.	78
Using the system does not require technical skills I do not have.	93
The level of the texts I had to read is good for me (not too hard and not too easy)	82
Using the system is an interesting experience for learning English.	79

Table 3 Subjects' questionnaire with the means of evaluations

Conclusion and Future work

I presented an intelligent tutoring system designed to optimize ESL learners' reading activities. The main aspect of this system is to use multiple features covering the major linguistic subdomains in the process of encoding both the difficulty of the texts provided by a corpus and the learner's proficiency.

The evaluation shows that the learners provided positive feedback about their experience with the system. Despite this positive feedback, there are several areas that are worth to explore for possible improvements.

First, involving a larger population of subjects in the evaluation would allow us to have a statistically representative pool for text sequence analysis. Consequently, this will help to do a deeper assessment of the strengths and weaknesses of the system.

Second, a consideration of the popularity of texts among other learners with similar profile could also be integrated.

References

Council of Europe, 2011. Common European framework of reference for languages: learning, teaching, assessment, online document, Retrieved November 20, 2017. https://www.coe.int/t/dg4/linguistic/source/framework_en.pdf

Guiraud, P. 1960. *Problèmes et Méthodes de la Statistique Linguistique.* D. Reidel, Dordrecht.

Harley, B.; King, M. L. 1989. Verb Lexis in the Written Compositions of Young L2 Learners, Volume 11, Issue 4 December 1989, pp. 415-439. https://doi.org/10.1017/S0272263100008421

Ure, J. 1971. Lexical density and register differentiation. In G. Perren and J.L.M. Trim (eds), *Applications of Linguistics*, London: Cambridge University Press. 443-452.

Zampa V.; Lemaire B. 2002. Latent Semantic Analysis for Student Modeling, *Journal of Intelligent Information Systems*, Special Issue on Education Applications 18(1), 15-30.

Kurdi, M. Zakaria, 2017. Lexical and Syntactic features selection for an adaptive reading recommendation system based on text complexity, International Conference ICISDM, Apr 1-3, Charleston, SC.